

## SMTP SMTP (Servidor Mega Tecnológico Perfeito SMTP)

---

Leon F. Bellini (22218002-8)  
Gulherme O. Sampaio (22218007-7)  
Felipe Maciel de Sousa (22218042-4)

---

Nenhum direito reservado.

# 1 Introdução

## 1.1 Execução

O *script* se encontra na raiz do projeto, nomeado simplesmente como `script.py`, para se executar em sistemas GNU/Linux, basta chamar o executável do **Python versão 3**.

Em distribuições baseadas em Debian, o comando deve incluir a versão específica do Python `python3`. Logo, o comando final seria `python3 script.py`.

O usuário será recebido, primeiramente, com um *prompt* para definir o número de servidores SMTP que deseja.

## 1.2 Dependências

O programa em si depende dos seguintes módulos/APIs para o seu funcionamento correto:

1. **requests**: Para a simulação do banco de domínios de um servidor DNS
2. **stmplib**: Para as funções/métodos relacionados ao envio e recebimento de e-mails para um determinado endereço IP.

Caso nenhuma dessas dependências estejam instaladas, basta utilizar-se do *package manager pip*, cuja ação para instalação de um pacote pode ser chamada com a opção `pip install`.

## 2 Mandando um e-mail

Como determinar a quantidade de servidores bem como os seus domínios de destinatário e remetente. Como determinar as portas destes servidores e como enviar o(s) e-mail(s).

### 2.1 Determinando a quantidade de servidores

O usuário pode inserir a quantidade qualquer de servidores os quais quiser instanciar, uma vez que haja memória disponível para tal ação.

### 2.2 Domínio e Porta

Após determinar a quantia de servidores que desejar, o usuário ainda pode determinar o “pseudo-domínio” para cada um deles (**É recomendado que não sejam utilizados caracteres que não sejam ASCII**). Estes serão, posteriormente, registrados no arquivo `dominios.txt`. Logo após digitar a *string* de domínio, o usuário deverá escolher uma porta que este servidor terá que reservar internamente. Se esta escolhida já estiver em uso pelo sistema/usuário, o programa entrará em estado de **falha**. É recomendada a utilização de uma porta não reservada ou que esteja ligada a um protocolo já existente como **ssh**.

O sucesso da instanciação do novo servidor será mostrado no *stdout* com o seguinte padrão:

```
Servidor rodando [porta <porta1>]
Servidor rodando [porta <porta2>]
.
.
.
```

### 2.3 Escrevendo o e-mail

Primeiramente será pedido um endereço de destinatário, **qualquer endereço é válido**, uma vez que o servidor de DNS retornará o IP para servidores nas redes externas, graças ao pacote **requests**. Esta regra vale tanto para o destinatário, quanto para o remetente. **Entretanto, o servidor externo pode negar qualquer tentativa de conexão, podendo, então levar a aplicação para o estado de erro.**

#### 2.3.1 Corpo e assunto da mensagem

Tanto o corpo quanto o assunto das mensagens do e-mail atual podem ser escritos com qualquer que o usuário desejar *charset*, como UTF-8.

#### 2.3.2 Anexos

O caminho do anexo a ser enviado **DEVE** ser absoluto, como:

```
/home/<usuario>/foto_de_casamento.png
```

Também é recomendado que não hajam espaços presentes no nome do arquivo. É necessário digitar 0 caso não se é desejado ter mais anexos.

## **2.4 Enviando o e-mail**

Após todos os passos terem sido seguidos corretamente, o e-mail será enviado para o destinatário desejado.

## 3 Considerações finais

### 3.1 O trabalho

O processo de criação dos servidores SMTP tomou mais ou menos duas semanas. A escrita dos códigos, entretanto, aconteceu de maneira gradual.

O grupo, de início, não compreendeu o objetivo final do projeto, uma vez que o enunciado em si não estava “solidificado” o bastante no primeiro momento em que foi passado. Assim que o projeto fora “oficializado”, foi possível entender como o processo deveria ocorrer. É recomendado ao professor que haja uma listagem de todos os requisitos do projeto na primeira vez que for passado um enunciado como esse.

Contanto, o desenvolvimento do projeto, bem como sua documentação fora divertida e permitiu um aprendizado melhor do protocolo SMTP, gerando a oportunidade de compreender todo o processo enquanto era praticada a programação deste.